

# A Fast Pairwise Evaluation of Molecular Surface Area

VLADISLAV VASILYEV, ENRICO O. PURISIMA

Biotechnology Research Institute, National Research Council of Canada, 6100 Royalmount Avenue, Montreal, Quebec, H4P 2R2, Canada

Received 7 March 2001; Accepted 2 October 2001

DOI 10.1002/jcc.10035

**Abstract:** A fast and general analytical approach was developed for the calculation of the approximate van der Waals and solvent-accessible surface areas. The method is based on three basic ideas: the use of the Lorentz transformation formula, a rigid-geometry approximation, and a single fitting parameter that can be refitted on the fly during a simulation. The Lorentz transformation equation is used for the summation of the areas of an atom buried by its neighboring contacting atoms, and implies that a sum of the buried pairwise areas cannot be larger than the surface area of the isolated spherical atom itself. In a rigid-geometry approximation we numerically calculate and keep constant the surface of each atom buried by the atoms involved in 1-2 and 1-3 interactions. Only the contributions from the nonbonded atoms (1-4 and higher interactions) are considered in terms of the pairwise approximation. The accuracy and speed of the method is competitive with other pairwise algorithms. A major strength of the method is the ease of parametrization.

© 2002 Wiley Periodicals, Inc. J Comput Chem 23: 737–745, 2002

**Key words:** analytical surface areas; derivatives; Lorentz formula; pairwise approach; solvent-accessible surface area (SASA); van der Waals surface area (vdWSA)

## Introduction

The solvent-accessible surface of a molecule is the surface generated by the center of solvent probe sphere rolled over the van der Waals surface.<sup>1</sup> Solvation-free energies of nonpolar molecules are observed to be roughly proportional to the solvent-accessible surface area (SASA) of the molecules.<sup>2</sup> As a result, a number of approaches based on this assumption have been devised.<sup>3–8</sup> The SASA can give us important information about the solvation properties of the molecules, and it would be very desirable to use it for molecular dynamics, Monte Carlo, or minimization techniques without explicit solvent molecules.

Usually the nonelectrostatic contribution to the solvation energy is described by linear functions of the total SASA

$$E = \lambda \cdot \text{Area} + b \quad (1)$$

or as a sum of linear functions for  $n$  different atom types

$$E = \sum_{i=1}^n \alpha_i \cdot \text{Area}_i + b \quad (2)$$

where the parameters  $\alpha_i$ ,  $\lambda$ , and  $b$  are parameters fitted to the experimental data.

From the point of view of the computational efficiency, it would be very attractive to approximate the molecular surface by the sum of the pairwise interactions of atoms in the molecule. If, in addition, a functional form of the resulting pairwise approach would be easily differentiable then it could be effectively used in molecular dynamics and optimization techniques. A number of authors have described such fast approximate pairwise methods for computing the SASA and van der Waals surface areas (vdWSA) of the molecules.<sup>9–13</sup> In this article we present a novel approximate pairwise approach for calculating the vdWSA and SASA that is simple to calculate and easy to parametrize.

To perform molecular dynamics simulations or energy minimization with energy functions containing surface area-based solvation terms, we need to be able to calculate the derivative of the SASA with respect to each of the atomic coordinates. Hence, it would be advantageous for any approximate estimate of the SASA to have an easily calculable derivative.

In general, for many implementations of Molecular Dynamics and optimization procedures, it is more important to reproduce the change in SASAs rather than the absolute magnitude of the SASA itself. The same is true even for Monte Carlo approaches or

**Correspondence to:** E. O. Purisima; e-mail: Enrico.Purisima@nrc.ca

Contract/grant sponsor: National Research Council of Canada; publication number 44808

optimization techniques that do not use derivatives. We take this into account in deciding whether to fit to absolute SASA or to relative changes in SASA for the parametrization of our approximate method.

## Method

Our approach is based on three basic ideas: a rigid-geometry approximation, the use of the Lorentz transformation formula, and an adaptive fitting procedure.

### Rigid-Geometry Approximation

The surface area buried from 1-2 and 1-3 interactions are the most difficult to calculate accurately using pairwise functions due to the significant multiple overlaps of the buried regions. This problem can be circumvented by taking a rigid-geometry approximation, i.e., bond lengths and angles will not vary much during the course of the simulation. With this assumption, the contribution from 1-2 and 1-3 interactions can be calculated once using a precise method at the start of the simulation and used throughout thereafter. In this work we use the fast and accurate numerical method of Bliznyuk and Gready<sup>14,15</sup> to calculate the surface buried by the 1-2 and 1-3 interactions. It is, of course, possible to update these areas periodically in the course of the simulation. The 1-4 and higher contributions (hereafter referred to as  $1 \geq 4$ ) are calculated using a pairwise function as described next.

### The Use of the Lorentz Transformation Formula

In the pairwise approximation of surface areas, one typically calculates for each atom the surface area buried by its neighboring atoms and then subtracts the sum of these areas from the theoretical area of the nonoccluded (spherical) atom to obtain its exposed surface area. One of the difficulties encountered in this approximation is that the sum of the buried areas from pairwise interactions can exceed the total free area of the atom in question. This is because, even for  $1 \geq 4$  interactions, the areas buried by neighboring atoms are not exclusive of one another and result in multiple counting of buried regions. One way to overcome this is to use the functional form of the Lorentz transformation from special relativity.<sup>16</sup> In relativity, this formula limits the sum of two velocities to be less than or equal to the speed of light. We will use the formula here to ensure that the sum of two or more buried surface areas will not exceed the total free area of an atom. Let  $A_{i,0}$  be the area of atom  $i$  that is buried by its 1-2 and 1-3 neighbors. If we now add to this the buried area due to atom 1 in the list of  $1 \geq 4$  neighbors of atom  $i$ , the "Lorentz formula" for combined area,  $A_{i,1}$ , is

$$A_{i,1} = \frac{A_{i,0} + B_{i,1}}{1 + \frac{A_{i,0}B_{i,1}}{S_i^2}} \quad (3)$$

where  $B_{i,1}$  is the added buried area and  $S_i$  is the area of an isolated atom  $i$ . The main characteristic of eq. (3) is that  $A_{i,1}$  is guaranteed

to be less than or equal to  $S_i$ . We can define a recursive expression for the sequential addition of buried areas from the other neighbors of atom  $i$ :

$$A_{i,j} = \frac{A_{i,j-1} + B_{i,j}}{1 + \frac{A_{i,j-1}B_{i,j}}{S_i^2}} \quad (4)$$

where  $A_{i,j-1}$  is the combined buried surface area of atom  $i$  due to 1-2, 1-3 neighbors and the first  $j - 1$   $1 \geq 4$  neighbors.  $B_{i,j}$  is the area that is buried by atom  $j$ . The total buried surface area of atom  $i$  is then given by  $A_{i,n}$ , where  $n$  is the number of  $1 \geq 4$  neighbors of atom  $i$ . The total exposed surface of the molecule is then

$$S = \sum_{i=1}^N (S_i - A_{i,n}) \quad (5)$$

where  $N$  is the total number of atoms in the molecule.

The term  $B_{i,j}$  in eq. (4) is calculated as

$$B_{i,j} = f \cdot \pi R_i (R_i + R_j - r_{ij}) \left( 1 + \frac{R_j - R_i}{r_{ij}} \right) \quad (6)$$

where  $R_i$  and  $R_j$  are the atomic radii of atoms  $i$  and  $j$ , respectively, and  $r_{ij}$  is the interatomic distance. The factor,  $f$ , is a fitting parameter to compensate for the overlaps of the buried regions.

The calculation of the derivatives is presented in Appendix.

### Fitting Strategy

For a given calibration set of  $N$  molecules, the adjustable parameter,  $f$ , was fitted by minimizing the relative error compared to an accurately computed area:

$$Err(f) = \frac{1}{N} \sum_{i=1}^N \frac{\text{abs}(S_{\text{ref}}^i - S_{\text{calc}}^i(f))}{S_{\text{ref}}^i} \quad (7)$$

The calibration set may consist of different molecules or various conformations of the same molecule.  $S_{\text{ref}}^i$  is the area calculated using the accurate numerical method of Bliznyuk and Gready<sup>14,15</sup> using atomic spheres with 1024 surface points each.

In many cases we are more interested in the change in the surface area rather than in the absolute value of the area itself. An alternate strategy then is to fit  $f$  to best reproduce the difference in surface areas between all pairs of molecules in the calibration set. Minimizing the following function will accomplish this.

$$Err(f) = \frac{1}{M} \sum_{i=1}^{N-1} \sum_{j=i+1}^N \text{abs}(\Delta S_{\text{ref}}^{ij} - \Delta S_{\text{ref}}^{ij}(f)) \quad (8)$$

where

$$\Delta S_{\text{ref}}^{ij} = S_{\text{ref}}^j - S_{\text{ref}}^i \quad (9)$$

$$\Delta S_{\text{calc}}^{ij}(f) = S_{\text{calc}}^i(f) - S_{\text{calc}}^j(f) \quad (10)$$

$$M = N(N - 1)/2 \quad (11)$$

Here,  $N$  is the number of different molecules or the number of different conformations of the same molecule. Fitting of the parameter  $f$  was carried out using a downhill Simplex method.<sup>17</sup>

It should be pointed out that by fitting  $f$  according to eq. (8), the total surface area calculated according to eq. (5) is no longer guaranteed to be accurate because the fitting was done on relative differences in area rather than on the absolute area. To put it another way, fitting to the differences in SASA implicitly introduces a second additive parameter that is needed for calculating the absolute SASA. By this we mean that for a given calibration set, adding a constant  $c$  to the each SASA calculated using eq. (5) will not change the fit to the differences in SASA. However, it will most certainly affect the agreement with the absolute SASA. Hence, to recover the absolute SASA using a value of  $f$  obtained using eq. (8) we have to augment the calculated SASA by an additive constant  $c$ .

$$S_{\text{abs}} = S + c \quad (12)$$

The constant,  $c$ , is readily obtained as the negative of the average deviation of the SASAs calculated using eq. (5) from the true values in the calibration set.

$$c = -\frac{1}{N} \sum_{i=1}^N (S_{\text{calc}}^i - S_{\text{ref}}^i) \quad (13)$$

### Test Set

Our test set included 10 polypeptide molecules ranging in size from alanine to penicillopepsin (2366 nonhydrogen atoms). These compounds are Nme-Ala-Ace, peptides P-6, P-12, and P-23 consisting of 6, 12, and 23 aminoacids, respectively, 1crn (crambin), 2ins (bovine insulin), 1lz1 (human lysozyme), 1inc (porcine elastase), 1kvd (toxin from halotolerant yeast), 3app (penicillopepsin). Peptides P-6, P-12, and P-23 were prepared in two different conformations—fully extended and folded ones. Coordinates of 1crn, 2ins, 1lz1, 1inc, 1kvd, 3app were taken from the protein data bank.<sup>18</sup>

To test the flexibility of the method we used two radius sets. The first one (JCC98) includes only nonhydrogen atoms:<sup>19</sup> C 1.70 Å, N 1.65 Å, O 1.60 Å, S 1.90 Å, i.e., only heavy atoms have nonzero radii. The second set (Amber95) also includes only heavy atoms in molecules and atomic radii were equal to the corresponding van der Waals radii from the Amber95 force field.<sup>20</sup> The atomic radii of all the hydrogens were set to zero. Eliminating hydrogen atoms, which constitute about half of the total number of atoms in proteins, leads to a drastic reduction of the number of the interatomic intersections (more than 2.5 times in the case of SASAs).

## Results and Discussion

### Van Der Waals Surfaces

Although the approximate calculation of the vdWSA has not been our primary objective, we took this as our first test case. We calibrated  $f$  using eq. (7) for both JCC98 and Amber95 radii sets (see Table 1). The calibration set consisted of all 10 molecules shown in the table. In all the cases there is a fairly good agreement between the numerical and pairwise vdWSA, with the total relative errors being 0.77 and 2.45% for the JCC98 and Amber95 radii set, respectively. The larger error in the case of the Amber95 radii set arises from the larger values of the Amber95 vdW radii. As a result, this leads to a larger number of intersections per atomic sphere (see Table 1), and consequently, to a larger error. It is interesting to note that in the case of the JCC98 radii set our approximate method with only one adjustable parameter gives practically the same relative error for the vdWSAs as does (ca. 0.8%) a more elaborate pairwise method, LCPO, (Linear Combination of Pairwise Overlaps), which uses four fitting parameters for every atom type (21 atom types in total).<sup>13</sup> This is probably due to the low number of  $1 \geq 4$  overlaps that have to be calculated using the approximate pairwise function.

Table 1 also contains mean absolute errors (in Å<sup>2</sup>) for the individual atomic vdWSAs, with the errors being 0.63 and 1.05 Å<sup>2</sup> for the JCC98 and Amber95 radii sets, respectively. It is a bit larger than a corresponding error of 0.37 Å<sup>2</sup> in the LCPO approximation.<sup>13</sup> Scatter plots of the numerical vs. pairwise atomic vdWSAs for two different radii sets are shown in Figures 1a and 1b.

### Solvent-Accessible Surfaces

We fitted  $f$  for the SASA using eq. (7) on the calibration set. The results are summarized in Table 2. Unlike the vdWSA, we see a large variation in the accuracy of the calculated SASA with errors ranging from 0.5 to 25%. The reason is that while the average number of intersections per sphere was fairly constant for the vdWSA, it can vary from 2 to 22 for the SASA when we go from small molecules to folded proteins. As a consequence, the use of a single global value for  $f$  for all the differently sized molecules can lead to large errors for some of the molecules. Scatter plots of the numerical vs. pairwise atomic SASAs for the two different radii sets are shown in Figures 2a and 2b. We see that for the more highly exposed atoms, there is a tendency to underestimate the SASA. However, for the more buried atoms, the estimated SASAs are scattered more evenly about the diagonal. For atoms with nearly zero SASA, we will, in fact, tend to overestimate the SASA. For the total molecular surface there will be a partial cancellation of errors from these two effects. For example, the mean values of the signed error in the estimated vs. exact SASA as given by

$$\frac{1}{N} \sum_{i=1}^N (S_{\text{ref},i} - S_{\text{calc},i}) \quad (14)$$

where  $N$  is the total number of atoms in all the sample molecules with nonzero radii ( $N = 8902$  in our case), are  $-0.060$  and  $-0.063$  Å<sup>2</sup> for the JCC98 and Amber95 radii, respectively.

**Table 1.** vdWSA Calculations Using JCC98 and Amber95 Radii Sets.

Molecule	No. spheres <sup>a</sup>	JCC98 radii ( $f = 0.458$ )			Amber95 radii ( $f = 0.449$ )		
		Ave. no. intersections per sphere <sup>b</sup>	% error <sup>c</sup>	Mean abs. atomic error <sup>d</sup> ( $\text{\AA}^2$ )	Ave. no. intersections per sphere <sup>b</sup>	% error <sup>c</sup>	Mean abs. atomic error <sup>e</sup> ( $\text{\AA}^2$ )
Ala	10	0.50	0.90	0.54	0.80	3.52	0.71
P-6 unfolded	40	1.03	1.04	0.45	1.58	5.93	0.98
P-6 folded	40	1.18	1.24	0.57	1.55	4.08	0.90
P-12 unfolded	85	1.07	1.46	0.50	1.64	5.58	1.03
P-12 folded	85	1.19	1.14	0.48	1.94	2.24	0.99
P-23 unfolded	170	1.06	1.26	0.49	1.61	5.48	1.03
P-23 folded	170	1.18	0.54	0.51	1.94	1.35	0.92
1ern	327	1.24	0.49	0.67	2.21	0.55	1.09
2ins	770	1.12	0.01	0.60	1.98	0.03	0.97
1lz1	1029	1.17	0.74	0.67	2.13	1.41	1.08
1inc	1822	1.12	0.56	0.64	2.14	0.11	1.06
1kvd	1988	1.11	0.52	0.62	2.12	1.31	1.05
3app	2366	1.14	0.11	0.64	2.13	0.32	1.07
Total	8902		0.77	0.63		2.45	1.05

$f$  was obtained by the fitting against the whole set of molecules using eq. (7).

<sup>a</sup>Number of atoms with nonzero radii.

<sup>b</sup>Only intersections of atoms participating in  $1 \geq 4$  interactions were calculated.

<sup>c</sup>Calculated as  $100 \times \text{abs}(S_{\text{ref}} - S_{\text{calc}})/S_{\text{ref}}$  where  $S_{\text{ref}}$  is calculated using the method of Bliznyuk and Gready.<sup>14</sup>

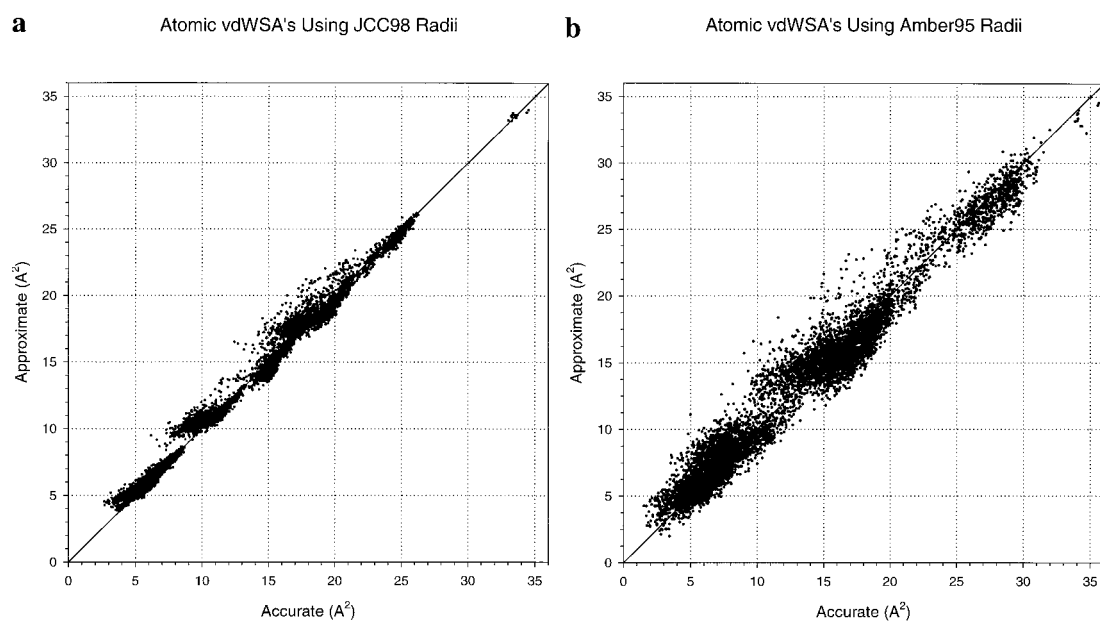
<sup>d</sup>Atomic surface areas range from 2.7 to 34.5  $\text{\AA}^2$ .

<sup>e</sup>Atomic surface areas range from 1.5 to 35.6  $\text{\AA}^2$ .

### Molecule-Specific Fitting

In this section we investigate the range of values of  $f$  obtained when fitting specifically to a given molecule. An equally important question is how well a single value of  $f$  for a given

molecule will perform for different conformations of that molecule. This becomes quite relevant for applications involving molecular dynamics simulations or energy minimization calculations. To explore these issues we subjected each of the mol-



**Figure 1.** Scatter plots of the approximate vs. accurate atomic surface area (in  $\text{\AA}^2$ ) for all the atoms in the data set. (a) vdW atomic surface areas using JCC98 radii set;  $r^2 = 0.984$ ; (b) vdW atomic surface area using Amber95 radii set;  $r^2 = 0.963$ .

**Table 2.** SASA Calculations Using JCC98 and Amber95 Radii Sets.

Molecule	JCC98 radii ( $f = 0.244$ )				Amber95 radii ( $f = 0.205$ )		
	No. spheres <sup>a</sup>	Ave. no. intersections per sphere <sup>b</sup>	% error <sup>c</sup>	Mean abs. atomic error <sup>d</sup> ( $\text{\AA}^2$ )	Ave. no. intersections per sphere <sup>b</sup>	% error <sup>c</sup>	Mean abs. atomic error <sup>e</sup> ( $\text{\AA}^2$ )
Ala	10	2.30	15.17	6.24	2.30	14.87	5.62
P-6 unfolded	40	5.90	24.62	5.86	6.50	22.85	5.78
P-6 folded	40	7.80	26.79	6.92	8.55	23.89	6.88
P-12 unfolded	85	6.72	24.78	5.54	7.55	22.94	5.48
P-12 folded	85	11.15	26.28	5.30	12.47	24.57	5.43
P-23 unfolded	170	6.99	25.02	5.29	7.91	22.96	5.17
P-23 folded	170	13.63	21.94	4.62	15.52	20.64	4.84
1ern	327	16.52	18.04	4.02	19.09	17.28	4.27
2ins	770	16.94	1.26	3.77	19.86	2.88	3.95
1lz1	1029	18.12	0.57	3.68	21.22	0.40	3.84
1inc	1822	18.55	7.67	3.55	21.92	9.21	3.65
1kvd	1988	18.91	8.81	3.46	22.19	9.34	3.55
3app	2366	18.87	10.71	3.67	22.29	10.93	3.74
Total	8902		16.28	3.73		15.60	3.83

$f$  was obtained by the fitting against the whole set of molecules using eq. (7).

<sup>a</sup>Number of atoms with nonzero radii.

<sup>b</sup>Only intersections of atoms participating in  $1 \geq 4$  interactions were calculated.

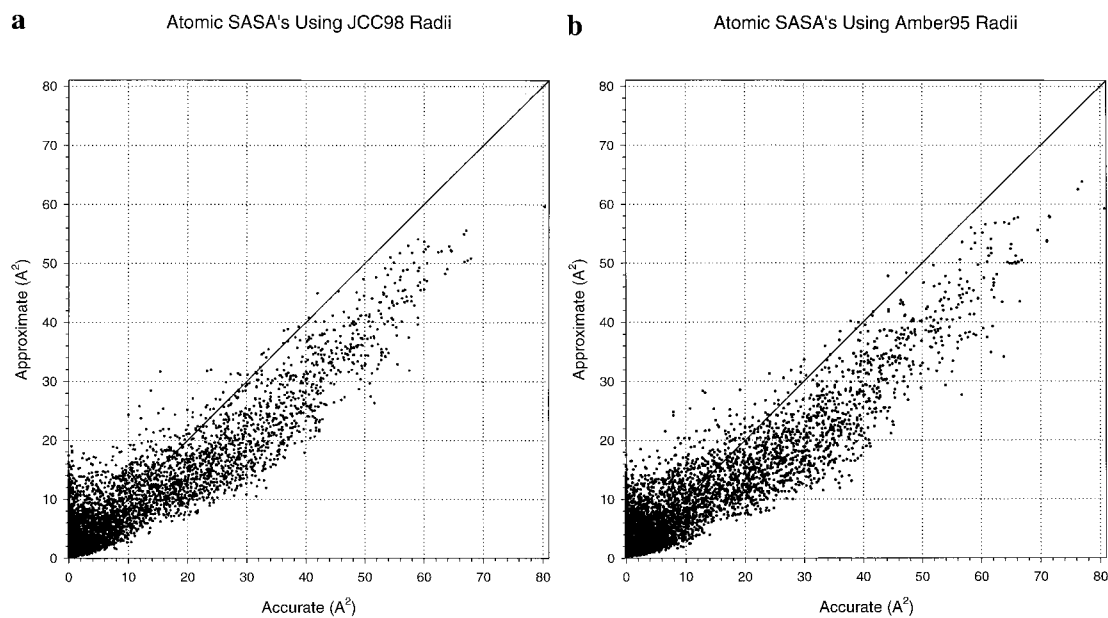
<sup>c</sup>Calculated as  $100 \times \text{abs}(S_{\text{ref}} - S_{\text{calc}})/S_{\text{ref}}$ , where  $S_{\text{ref}}$  is calculated using the method of Bliznyuk and Gready.<sup>14</sup>

<sup>d</sup>Atomic surface areas range from 0 to  $80.3 \text{ \AA}^2$ .

<sup>e</sup>Atomic surface areas range from 0 to  $80.7 \text{ \AA}^2$ .

ecules in the original calibration set to a short molecular dynamics run and took snapshots at 200-fs intervals to create a new calibration set for each molecule consisting of the snap-

shots. We examined the two strategies of fitting to the SASA, using eq. (7), or to the differences in the SASA among the conformations, eq. (8).



**Figure 2.** Scatter plots of the approximate vs. accurate atomic surface areas (in  $\text{\AA}^2$ ) for all the atoms in the data set. The accurate values were calculated using the method of Bliznyuk and Gready.<sup>14</sup> (a) Solvent-accessible atomic surface area using JCC98 radii set;  $r^2 = 0.859$ ; (b) solvent-accessible atomic surface area using Amber95 radii set;  $r^2 = 0.847$ .

**Table 3.** Molecule-Specific Parameters Applied to Conformations Generated by an MD Run.

Molecule	Fit to absolute SASA <sup>a</sup>			Fit to change in SASA <sup>b</sup>				Single $f$ parameter <sup>c</sup> ( $f = 0.205$ )		
	$f$	% error	Eq. (8), Å <sup>2</sup>	$f$	$c$ (Å <sup>2</sup> )	% error	Eq. (8), Å <sup>2</sup>	$c$ (Å <sup>2</sup> )	% error	Eq. (8), Å <sup>2</sup>
Ala	0.094	0.5	2	0.075	-10	0.4	2	53	0.6	3
P-6 unfolded <sup>c</sup>	0.117	0.6	7	0.217	237	0.5	6	211	0.5	6
P-6 folded <sup>c</sup>	0.129	0.9	11	0.236	271	0.8	10	203	0.2	10
P-12 unfolded <sup>c</sup>	0.143	2.2	49	0.200	317	2.2	44	339	2.2	44
P-12 folded <sup>c</sup>	0.153	1.5	26	0.236	409	1.4	24	279	1.4	24
P-23 unfolded <sup>c</sup>	0.161	5.2	193	0.196	442	5.5	188	523	5.6	189
P-23 folded <sup>c</sup>	0.159	5.2	194	0.296	1015	1.0	30	-392	1.1	32
1crn	0.180	0.8	37	0.417	2215	0.6	28	-413	0.8	36
2ins	0.209	1.2	97	0.394	3672	0.7	63	181	1.2	98
1lz1	0.216	1.0	102	0.383	4226	0.6	59	470	1.1	107
1inc	0.230	0.8	125	0.310	4061	0.7	113	1840	0.9	133
1kvd	0.228	0.7	109	0.284	3431	0.5	85	1941	0.8	129
3app	0.228	0.9	161	0.369	7651	0.4	73	2337	1	183

Starting conformations for the MD runs were the structures used in Tables 1 and 2. The MD calculation was carried out using the Amber force field at 300 K, a 1-fs time step, a 10-Å cutoff, and dielectric function of 4 $r$ ; 50–100 snapshots taken at 200-fs intervals for each molecule were used for the fitting. Amber95 radii were used for the SASA calculation.

<sup>a</sup>The parameter  $f$  and % error were calculated by minimizing eq. (7).

<sup>b</sup>The parameter  $f$  was obtained by minimizing eq. (8) and  $c$  was fitted using eq. (13). The % error is given by eq. (7) with  $S_{\text{calc}}$  calculated using eq. (12).

<sup>c</sup> $f$  was fixed at 0.205 and  $c$  was fitted using eq. (13).

The results are summarized in Table 3 where we tabulate the deviations of the calculated SASA from the true values. For comparison, we also present the results using the global value of  $f$  obtained previously from the original calibration set. We note the marked reduction in error in the calculated SASA when using molecule-specific  $f$  values. The majority of the calculated areas are within about 1% of the true value. Even the worst case is within 5% of the true value. We also see that for a given molecule, a single value of  $f$  suffices for the range of conformations produced during a 10–20-ps MD run. This suggests an adaptive fitting strategy for simulations that use our pairwise approximation of the SASA as part of the energy function. This involves initially fitting the parameter  $f$  using the starting structure after which the simulation is then run with the fitted  $f$  for a given period of time or until the calculated SASA changes significantly (suggesting a major conformational change). The parameter  $f$  can then be refitted from snapshots using the most recent conformations visited in the MD run and the MD continued. This process can then be repeated cyclically to have an  $f$  optimized for the most recent conformations sampled in the MD run.

As discussed earlier in the Methods section, an alternative approach to fitting  $f$  is to fit to the pairwise differences in SASA among entries in the validation set followed by fitting a translation constant,  $c$ , for the absolute SASA. We examined this way of fitting using the MD snapshots for the test molecules. The results are summarized in Table 3. We note that the value of  $f$  obtained in this manner can be quite different from the single-parameter fit case. Yet, there is no significant difference in the relative error in the total surface area between the two ways of fitting  $f$ . This suggests that the method is fairly robust with respect to variations in  $f$  as long as the additive constant  $c$  is adjusted correspondingly.

This is seen more clearly in the last three columns of Table 3 where the value of  $f$  was arbitrarily set to a value of 0.205 and the constant  $c$  fitted for each molecule. We see that the error in the calculated SASA is comparable to the first two sets of results in Table 3.

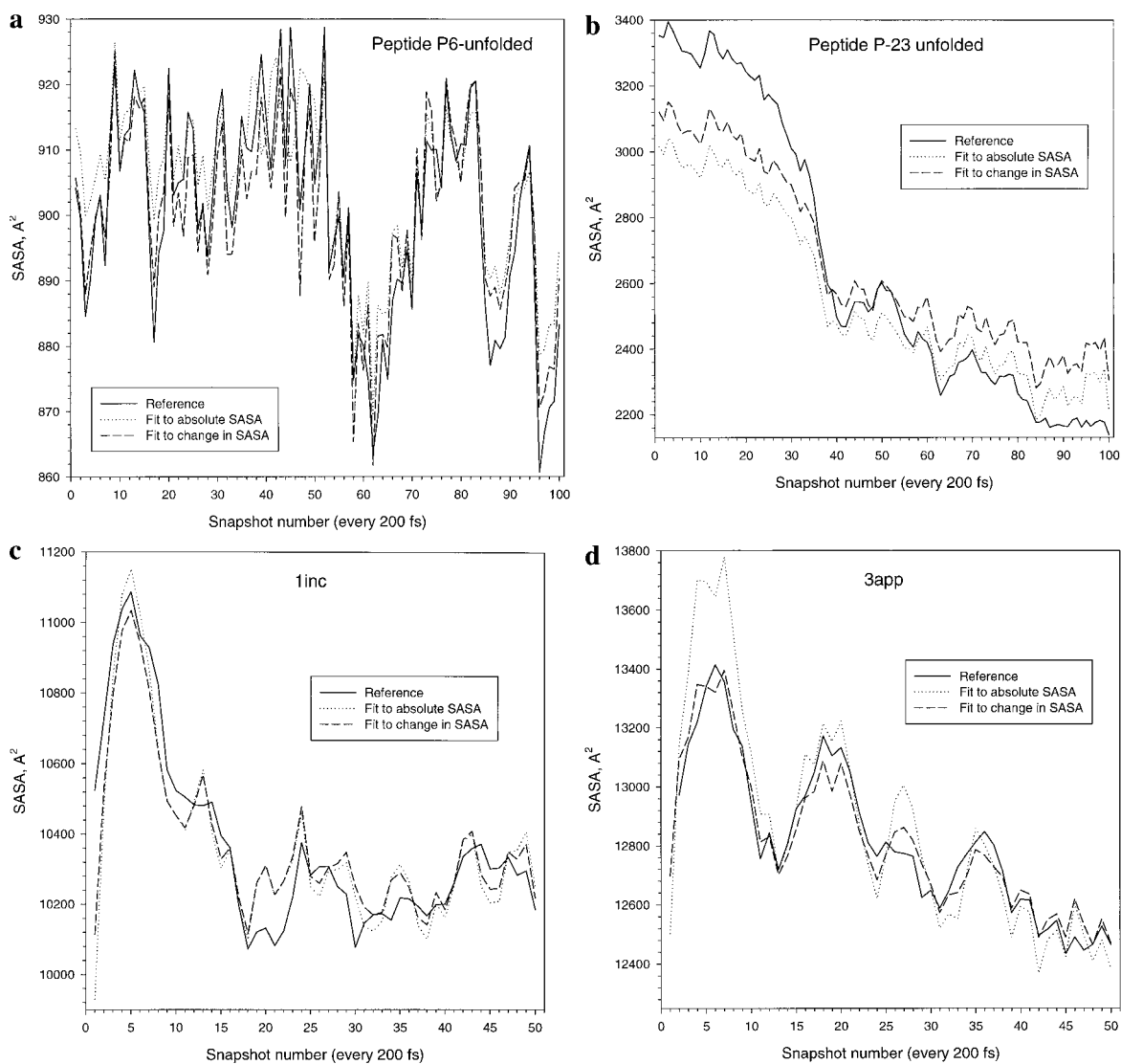
The advantage of fitting to changes in SASA becomes visible when we examine the average error of the calculated change in SASA between pairs of conformations. For some of the molecules (e.g., 1lz1, 3app) there is a significant reduction in this quantity when fitting specifically to the change in SASA. This suggests that for use in MD calculations fitting to the changes in SASA is preferable.

### SASA in Molecular Dynamics

To illustrate the dynamical behavior of the computed SASA we examine in more detail the SASA for the MD snapshots of four of the molecules in our test set: P-6 unfolded, P-23 unfolded, and 1inc and 3app. These examples range from unstructured polypeptides to folded proteins.

Figure 3a shows a plot of the calculated SASAs along the MD trajectory snapshots for P-6 (unfolded). The curve corresponding to the fit to the change in SASA tracks the reference curve better than the curve with a parameter fitted to the absolute SASA. However, even in the latter case, the calculated SASAs follow the changes in the true value throughout the MD run reasonably well. This suggests that the calculation of changes in SASA is fairly robust to variations in the value of the parameter  $f$ .

P-23 (unfolded) is a challenging case because there is a large change in surface area as the peptide goes from a fully extended conformation to a more compact globular shape during the course of the MD simulation. Even in this case, the calculated SASAs



**Figure 3.** Comparison of the approximate vs. accurate SASAs along an MD trajectory. The MD runs are as described in Table 3. The reference values of the SASA were calculated using the method of Bliznyuk and Gready.<sup>14</sup> (a) Unfolded p-6 peptide; (b) unfolded p-23 peptide; (c) 1inc (porcine pancreatic elastase); (d) 3app (penicillopepsin).

track the changes in the true SASAs well (Fig. 3b). At around the 40th snapshot there is a large conformational change to a more compact structure. The fitted parameters seem to be a compromise between the more extended and more compact forms. In an adaptive fitting strategy, we would refit the parameters shortly after this conformational change.

1inc is a medium-sized globular protein of 240 amino acids. Figure 3c shows the evolution of the SASAs along the MD trajectory. In this case also, the calculated SASAs follow the true values closely. Figure 3d shows an example of a case, 3app, where fitting to the changes in SASA yields an MD profile of the SASA superior to that of fitting to the absolute SASA.

#### Comparison with Other Methods

The use of a single fitting parameter [plus a trivially fitted additional one if we use eq. (12)] makes the method easy and quick to calibrate. Fitting of the adjustable parameter  $f$  converges very rapidly. This means that in the course of a simulation the parameter can be recalibrated on the fly to take into account possible large conformational changes requiring a modification of the adjustable parameter for optimal fit. New molecule classes pose no problems either because no new atom-type parameters need to be defined as in other pairwise methods. We are able to accomplish this partly because of the rigid geometry approximation that we employ.

Table 4 (last column) gives us an indication about the magnitude of the error of the rigid geometry approximation. Inaccuracies in the 1-2 and 1-3 overlaps, due to unaccounted changes in bond lengths and angles during MD simulation, contribute less than 1% to the overall inaccuracy of the molecular SASA calculation. The contribution of 1-2 and 1-3 overlaps can also be recalculated periodically if increased accuracy is desired.

Additional advantages come from the rigid-geometry approximation that reduces the number of interatomic intersections to be calculated. The reduction is especially pronounced in the case of the vdWSA calculations, and may be as much as two to three times (see Table 4).

A direct comparison of the speed of our method to reported benchmarks in the literature is not easy because it depends on the computer architecture of the reference machine, compiler options, and, of course, on the implementation of the algorithm. Given that caveat, we compared the performance of our method against three other algorithms in the literature. First, we carried out a side-by-side test against the approximate pairwise method of Hasel et al.<sup>10</sup> and the numerical method of Bliznyuk and Gready.<sup>14,15</sup> The method of Hasel et al. is very simple, and was straightforward to implement locally. The code for the method of Bliznyuk and Gready has been made available by the authors. The benchmarks were performed on an SGI R10000/250 MHz processor using C++ code optimized at the `-mips4 -O3` level.

Table 5 shows the CPU times for the analytical and numerical SASA calculations using the Amber95 radii. One can see that our method requires about the same amount of time for the SASA calculation as the pairwise method of Hasel et al.<sup>10</sup> In both cases, the rate-limiting step is the calculation of the buried surfaces of the

**Table 4.** Statistics for 1-2 and 1-3 Intersections.

Molecule	JCC98		Amber95		% error <sup>b</sup>
	vdWSA <sup>a</sup>	SASA <sup>a</sup>	vdWSA <sup>a</sup>	SASA <sup>a</sup>	
Ala	5.00	1.87	3.50	1.87	0.71
P6-unfolded	3.32	1.40	2.51	1.37	0.57
P6-folded	3.02	1.30	2.52	1.28	0.72
P12-unfolded	3.21	1.35	2.43	1.31	0.50
P12-folded	2.99	1.21	2.21	1.19	0.36
P23-unfolded	3.28	1.35	2.49	1.31	0.48
P23-folded	3.03	1.18	2.23	1.16	0.26
1crn	2.98	1.15	2.10	1.13	0.80
2ins	3.16	1.14	2.21	1.12	0.14
1lz1	3.06	1.13	2.12	1.11	0.27
1inc	3.16	1.13	2.16	1.10	0.06
1kvd	3.16	1.13	2.11	1.11	0.23
3app	3.12	1.13	2.13	1.11	0.11

<sup>a</sup>The ratio between the total number of atomic intersections and the intersections of atoms participating only in the  $1 \geq 4$  interactions using different radii. Only atoms with nonzero radii were considered.

<sup>b</sup>Relative error of the SASA rigid geometry approximation along the MD trajectories using Amber95 radii. Calculated as  $100/N \sum_{i=1}^N |(S_{1-2,3}^0 - S_{1-2,3}^i)/S_{1-2,3}^0|$ , where  $S_{1-2,3}^0$  is the area buried by the 1-2 and 1-3 intersections in the starting structure.  $S_{1-2,3}^i$  is the same quantity at the  $i$ th snapshot in the MD simulation.  $N$  is the number of snapshots in the trajectory.

**Table 5.** CPU Times for the SASA Calculations Using Different Methods and Amber95 Radii.

Protein	No. spheres <sup>a</sup>	CPU (s)			
		This work	This work <sup>b</sup>	Hasel et al. <sup>10</sup>	Bliznyuk and Gready <sup>14,15</sup>
1crn	327	0.02	0.05	0.01	0.14
2ins	770	0.05	0.16	0.05	0.31
1lz1	1029	0.08	0.23	0.07	0.46
1inc	1822	0.16	0.44	0.16	0.83
1kvd	1988	0.16	0.48	0.17	0.93
3app	2366	0.21	0.57	0.20	1.11

<sup>a</sup>Number of atoms with nonzero radii.

<sup>b</sup>Area and first derivative.

overlapping atoms [see eq. (6)]. We also see that numerical algorithm Bliznyuk and Gready<sup>14,15</sup> is quite fast, and is consistent with their conclusion at the time that their algorithm “is at least as fast as the fastest competitor algorithm for the evaluation of the solvent accessible surface area.”<sup>15</sup> Our method, without the gradient calculation is about five times as fast as their method. With the calculation of the derivative, it is still about twice as fast. Their numerical method does not provide estimates of the gradient.

The LCPO pairwise method has been described recently in the literature.<sup>13</sup> We have not benchmarked the LCPO pairwise method ourselves, but the reported benchmark in the literature uses a machine and compilation options similar to what we have used. For the protein 3app with 2366 atoms (one of our test cases also) they report a CPU time of 0.87 s for the SASA and its derivative. Our method takes 0.57 s for this molecule (Table 5). This makes their algorithm slightly slower than ours for this test case. It should be noted, however, that they use a large number of fitted parameters in their calculation.

## Summary

In this article we presented a fast pairwise approximation to the SASA. One of the main attractions of the method is the use of essentially a single fitting parameter. The algorithm is particularly suited for MD or Monte Carlo simulations of a molecule. Analytical derivatives are provided by the algorithm. New molecule classes and atom types require no additional parameters, as may be required in other pairwise methods. The performance and accuracy is competitive with other methods in the literature.

## Appendix

### First Derivatives

From eq. (5), the first derivative of the surface area,  $S$ , with respect to the  $x$ -coordinate of atom  $i$  is given by

$$\frac{\partial S}{\partial x_i} = -\frac{\partial A_{i,n}}{\partial x_i} - \sum_{j \neq i}^N \frac{\partial A_{j,n}}{\partial x_i} \quad (\text{A1})$$

where the first term on the right side of the equation represents the derivative of the buried surface of atom  $i$  by other atoms with respect to the  $x$ -coordinate of atom  $i$ . The second term is the derivative of the buried surfaces of atoms other than  $i$ .  $N$  is the number of atoms. The expressions for the  $y$ - and  $z$ -coordinates are entirely analogous.

For the first term we have

$$\frac{\partial A_{i,n}}{\partial x_i} = \frac{S_i^2 \left[ (S_i^2 - B_{i,n}^2) \frac{\partial A_{i,n-1}}{\partial x_i} + (S_i^2 - A_{i,n-1}^2) \frac{\partial B_{i,n}}{\partial x_i} \right]}{(S_i^2 + A_{i,n-1} B_{i,n})^2} \quad (\text{A2})$$

...

$$\frac{\partial A_{i,1}}{\partial x_i} = \frac{S_i^2 (S_i^2 - A_{i,0}^2) \frac{\partial B_{i,1}}{\partial x_i}}{(S_i^2 + A_{i,0} B_{i,1})^2} \quad (\text{A3})$$

And from eq. (6) we have for each pair

$$\frac{\partial B_{i,j}}{\partial x_i} = \frac{f\pi R_j (R_i^2 - R_j^2 - r_{ij}^2) (x_i - x_j)}{r_{ij}^3} \quad (\text{A4})$$

We could calculate the second term in eq. (A1) just like the first term. However, we can avoid a recursive derivative by using a simple trick. Let  $A_{j,n'}$  be the area of atom  $j$  buried by all atoms except atom  $i$ . If we now add the area buried by atom  $i$  to get the total buried area, we have by eq. (4)

$$A_{j,n} = \frac{A_{j,n'} + B_{j,i}}{1 + \frac{A_{j,n'} B_{j,i}}{S_j^2}} \quad (\text{A5})$$

The key characteristic of this expression is that  $A_{j,n'}$  does not depend on the coordinates of atom  $i$ .  $A_{j,n'}$  can be easily computed as

$$A_{j,n'} = \frac{S_j^2 (B_{j,i} - A_{j,n})}{B_{j,i} A_{j,n} - S_j^2} \quad (\text{A6})$$

This then gives for the derivative

$$\frac{\partial A_{j,n}}{\partial x_i} = \frac{S_j^2 f (S_j^2 - A_{j,n}^2) \frac{\partial B_{j,i}}{\partial x_i}}{(S_j^2 + f A_{j,n'} B_{j,i})^2} \quad (\text{A7})$$

$$\frac{\partial B_{j,i}}{\partial x_i} = \frac{f\pi R_j (R_j^2 - R_i^2 - r_{ij}^2) (x_i - x_j)}{r_{ij}^3} \quad (\text{A8})$$

## References

1. Lee, B.; Richards, F. M. *J Mol Biol* 1971, 55, 379.
2. Hermann, R. B. *J Phys Chem* 1972, 76, 2754.
3. Eisenberg, D.; McLachlan, A. D. *Nature* 1986, 319, 199.
4. Stouten, P. F. W.; Frömmel, C.; Nakamura, H.; Sander, C. *Mol Simulat* 1993, 10, 97.
5. Privalov, P. L.; Makhatazde, G. I. *J Mol Biol* 1993, 232, 660.
6. Fraternali, F.; van Gunsteren, W. F. *J Mol Biol* 1996, 256, 939.
7. Augspurger, J. D.; Scheraga, H. A. *J Comp Chem* 1996, 17, 1549.
8. Lazaridis, T.; Karplus, M. *Science* 1997, 278, 1928.
9. Wodak, S. J.; Janin, J. *Proc Natl Acad Sci USA* 1980, 77, 1736.
10. Hasel, W.; Hendrickson, T. F.; Still, W. C. *Tetrahedron Comput Method* 1988, 1, 103.
11. Kurochkina, N.; Lee, B. *Protein Eng* 1995, 8, 437.
12. Street, A. G.; Mayo, S. L. *Folding Design* 1998, 3, 253.
13. Weiser, J.; Shenkin, P. S.; Still, W. C. *J Comp Chem* 1999, 20, 217.
14. Bliznyuk, A. A.; Gready, J. E. *J Comp Chem* 1996, 17, 962 (the program code is available from <http://anusf.anu.edu.au/~aab900/area.html>).
15. Bliznyuk, A. A.; Gready, J. E. *J Comp Chem* 1996, 17, 970.
16. Lorrain, P.; Corson, D. R. *Electromagnetic Fields and Waves*; W. H. Freeman and Co.: New York, 1970.
17. Nelder, J. A.; Mead, R. *Comput J* 1965, 7, 308.
18. Berman, H. M.; Westbrook, J.; Zukang, F.; Gilliland, G.; Bhat, T. N.; Weissig, H.; Shidyalov, I. N.; Bourne, P. E. *Nucleic Acids Res* 2001, 28, 235.
19. Weiser, J.; Weiser, A. A.; Shenkin, P. S.; Still, W. C. *J Comp Chem* 1998, 19, 797.
20. Cornell, W. D.; Cieplak, P.; Bayly, C. I.; Gould, I. R.; Merz, K. M., Jr.; Ferguson, D. M.; Spellmeyer, D. C.; Fox, T.; Caldwell, J. W.; Kollman, P. A. *J Am Chem Soc* 1995, 117, 5179.